

Attached hereto is a marked-up version of the changes made to the claims by the current Amendment. The attached is captioned “**Version With Markings to Show Changes Made.**”

Rejection Under 35 U.S.C. §103:

Claims 1-2, 4-8 and 10-11 were rejected under 35 U.S.C. §103 as allegedly being unpatentable over Katseff et al (U.S. ‘796, hereinafter “Katseff”) in view of Gramlich (U.S. ‘025). Applicant respectfully traverses this rejection.

In order to establish a prima facie case of obviousness, all of the claimed limitations must be taught or suggested by the prior art and there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings.

Applicant respectfully submits that the combination of Katseff and Gramlich fails to teach or suggest all of the claimed limitations. For example, Applicant submits that the combination fails to teach or suggest a dynamic server computer running a program for transforming data which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the computer network.

The final Office Action admits that Katseff does not teach this claimed feature. (See section 3, pages 2-3 of the Office Action.) However, the Office Action asserts that “Gramlich teaches a system using multiple servers to

employ a remote program for annotating, filtering and transforming data (Gramlich, col. 2, lines 60-63 and col. 5, line 17 - col. 6, line 57).” Applicant respectfully disagrees with this assessment of Gramlich. In fact, what is disclosed in Gramlich is one server (the proxy server 130) running one program (the AOP 114) which includes a number of sub-routines (such as insert, delete, Run_Program), each of which is hard-coded into AOP 114. (Note that even in the case of “Run_Program”, the program identified in the command is one which is “associated” with AOP 114 and is located on the proxy server 130). However, what Gramlich fails to disclose is “a dynamic proxy server computer being configured to run a program for transforming data, which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the network.”

With respect to the above claimed feature, the final Office Action states “Additionally, annotations are performed dynamically, based on specific document requests and various sources, ranging from multiple AOP’s or other web servers (See also Gramlich, col. 13, line 50-col. 14, line 18).” This portion and other portions of Gramlich merely emphasize the fact that the “transformation” being accomplished by Gramlich involves the annotation overlay proxy (AOP) searching other web servers for all annotation overlays possibly applicable. This passage does not teach that a transformation process itself being dynamically downloaded to a dynamic proxy server. Instead, this passage merely explains that the AOP protocol (i.e., the annotation overlay

proxy protocol) searches the web for additional annotation data to be added to the protocol process being transformed by the AOP. The protocol defining the AOP process itself is represented by reference numeral 114 in Fig. 1 as being resident in the proxy server 130. There is no teaching or suggestion in Gramlich that the AOP 114 resident in proxy server 130 found its way there by dynamic downloading.

Gramlich teaches a method of adding one or more annotation overlays to a document stored on a web server (see Figs. 4 and 5 in particular), with the annotation overlays also being stored on a web server (although not necessarily the same web server) as that which stores the document. This allows a user to see a document, along with the comments of other users, even though the documents and the associated comments are not stored together. The document and the various annotations are loaded from their respective servers and combined by an annotation overlay proxy in response to a web browser wishing to see the document with the overlays. The “merged” annotated document is then served to the web browser by the AOP.

It is thus clear that the AOP acts to alter the content of the requested document, i.e., by adding the contents of the various annotation overlays to the requested document. This is in marked contrast with the claimed invention which recites that the dynamic proxy server computer transforms the data from a first encoding format to a second encoding format **without changing the information content** while the encoding format used to deliver the data is changed as appropriate. Gramlich teaches away from the claimed invention by

teaching that annotation overlays are added as HTML-formatted content to HTML documents so that the annotations are “seamlessly integrated” within the document (see col. 6, lines 6-24).

Claims 4 and 7 further require “the transforming performed by the dynamic proxy server computer is determined by the content of a protocol dynamically downloaded from a third server computer.” Applicant submits that neither Katseff nor Gramlich teaches or suggests this feature. The Office Action relies upon, for example, col. 5, lines 41-51 for a teaching of this limitation. However, this portion merely echoes that the AOP protocol searches the web for additional annotation data to be added to the protocol process being performed by the AOP. This passage does not teach that a transformation process itself is determined by the content of a protocol which is dynamically downloaded from a third server.

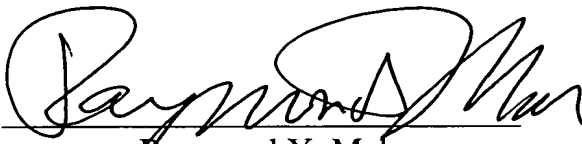
Applicant therefore submits that claims 1-2, 4-8 and 10-11 are not obvious over Katseff and Gramlich and respectfully requests that the rejection of these claims under 35 U.S.C. §103 be withdrawn.

Conclusion:

Applicant believes that this entire application is in condition for allowance and respectfully requests a notice to this effect. If the Examiner has any questions or believes that an interview would further prosecution of this application, the Examiner is invited to telephone the undersigned.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: 
Raymond Y. Mah
Reg. No. 41,426

RYM/sl
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
Telephone: (703)816-4044
Facsimile: (703)816-4100

VERSION WITH MARKINGS TO SHOW CHANGES MADE

IN THE CLAIMS:

1. (Thrice Amended) A method of transmitting data from a first server computer to a client computer via a computer network, the method comprising:

requesting a [second] dynamic proxy server computer to run a program for transforming data, which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the computer network;

requesting the first server computer to transmit the data to said [second] dynamic proxy server computer;

transmitting data from the first server computer to the [second] dynamic proxy server computer in a first encoding format,

transforming said data from said first encoding format to a second encoding format at the [second] dynamic proxy server computer without substantially changing the information content of said data; and

transmitting the transformed said data in said second encoding format to the client computer [form] from the [second] dynamic proxy server computer.

2. (Twice Amended) A method as in claim 1, wherein:

said data is transmitted from the first server computer to the [second] dynamic proxy server computer using a first transport protocol and the

transformed data is transmitted from the [second] dynamic proxy server computer to the client computer using a second transport protocol.

4. (Thrice Amended) A method of transmitting data from a first server computer to a client computer via a computer network, the method comprising:

requesting a dynamic proxy server computer to run a program for transforming data, which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the computer network;

requesting the first server computer to transmit the data to said dynamic proxy server computer;

transmitting data from the first server computer to the dynamic proxy server computer in a first encoding format,

transforming said data from said first encoding format to a second encoding format at the dynamic proxy server computer without substantially changing the information content of said data; and

transmitting the transformed said data in said second encoding format to the client computer from the dynamic proxy server computer;

[A method as in claim 1 wherein:] the transforming performed by the [second] dynamic proxy server computer is determined by the content of a

protocol dynamically downloaded to the [second] dynamic proxy server computer from a third server computer.

6. (Thrice Amended) A dynamic proxy server computer,
said dynamic proxy server computer being located in a computer network such that it is in a communications route intermediate a server computer and a client computer,
said dynamic proxy server computer being configured to run a program for transforming data, which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the computer network;
the dynamic proxy server computer being configured to receive data transmitted in a first data format from said server computer, to transform received data to a second data format from said first data format without substantially changing the information content of said data and to transmit the transformed said data to the client computer in said second data format.

7. (Twice Amended) A dynamic proxy server computer,
said dynamic proxy server computer being located in a computer network such that it is in a communications route intermediate a server computer and a client computer,

said dynamic proxy server computer being configured to run a program for transforming data, which program is referred to by an address within a data store connected to the computer network from where the computer program is available for dynamic downloading by server computers connected to the computer network;

the dynamic proxy server computer being configured to receive data transmitted in a first data format from said server computer, to transform received data to a second data format from said first data format without substantially changing the information content of said data and to transmit the transformed said data to the client computer in said second data format;

[A dynamic proxy server computer as in claim 6, wherein:]

the transforming performed by the dynamic proxy server computer is determined by the content of a protocol dynamically downloaded from a third server computer.

but this

11. (Thrice Amended) A computer network comprising:

a World Wide Web server,

a client computer, and

at least one dynamic proxy server computer,

the dynamic proxy server computer being located between the World Wide Web server and the client computer,

said dynamic proxy server computer being configured to run a program for transforming data, which program is referred to by an address within a data

store connected to the computer network from where the computer program is available for dynamically downloading by server computers connected to the computer network;

the dynamic proxy server computer being arranged to transform data transmitted from the World Wide Web server to the client computer from one format to another without substantially changing the information content of said data.